



Ishak, M. K., Herrmann, G., & Pearson, M. (2016). Performance evaluation using Markov model for a novel approach in Ethernet based embedded networked control communication. In *2016 Annual IEEE Systems Conference (SysCon 2016): Proceedings of a meeting held 18-21 April 2016, Orlando, Florida, USA* [7490584] Institute of Electrical and Electronics Engineers (IEEE).
<https://doi.org/10.1109/SYSCON.2016.7490584>

Peer reviewed version

License (if available):
Unspecified

Link to published version (if available):
[10.1109/SYSCON.2016.7490584](https://doi.org/10.1109/SYSCON.2016.7490584)

[Link to publication record in Explore Bristol Research](#)
PDF-document

This is the author accepted manuscript (AAM). The final published version (version of record) is available online via IEEE at <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7490584>. Please refer to any applicable terms of use of the publisher.

University of Bristol - Explore Bristol Research

General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:
<http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/>

Performance Evaluation Using Markov Model For A Novel Approach In Ethernet Based Embedded Networked Control Communication

Mohamad Khairi Ishak*, Guido Herrmann**, Martin Pearson***

* School of Electrical and Electronic Engineering, Universiti Sains Malaysia

Nibong Tebal, Penang, Malaysia.

**Queens School of Engineering

University Walk, Bristol BS8 1TR, United Kingdom

***Bristol Robotics Laboratory, Bristol, United Kingdom

khairiishak@usm.my, g.herrmann@bristol.ac.uk, martin.pearson@brl.ac.uk

Abstract—In earlier work, we suggested a low cost control communication scheme across half-duplex Ethernet allowing for reduced jitter and delay for an information sharing control communication bus. The principle advantage of half-duplex Ethernet is the reduced number of wires (two versus four wires) and the reduced cost of switch/router equipment. The basic principles for our communication has been the network wide introduction of a synchronisation signal and the modification of the back-off scheme for each node, by assigning to each a node specific minimal back-off time. In this paper, a Markov chain model for modeling network jitter and delays is developed for this control communication approach. For this, the events occurring after a synchronization signal are formally summarized within a state vector. A two node example is considered. The state vector describes the states of collision detection, back-off and transmission within the two node system. Relevant probabilities are assigned for transfer between different states. The process of calculating the overall Markov chain model and related parameters is described. This approach shows the advantage of our approach over a standard Carrier Sense Multiple Access/Collision Detection (CSMA/CD) setting.

Keywords—Markov Model, Backoff time, Jitter, Delays, Ethernet, and network model.

I. INTRODUCTION

For the last few decades, the increasing usage of electronics and computerised systems significantly impacted on the industry, automotive systems, avionics and robotics. In order to achieve a given task, these engineering systems will share sensor and actuator information in a distributed sense, e.g. torque, velocity, current and position. In this scenario, the communication between the technological systems and its sub-units is one of the main factors which contribute to the capabilities and effectiveness. Hence, the applications of embedded systems should timely deliver synchronized data-sets, minimize latency in their response and meet their performance target. Networked control communication is a backbone to accomplish tasks to be carried out for distributed dynamic control using embedded systems. With that, there are various types of communication protocols such as Controller Area Network (CAN), CANopen, Flexray, Time Triggered CAN

(TTCAN) and Time Triggered Ethernet (TTEthernet) [1], [2], [3], [4].

Modern wired, high speed Ethernet based systems (Profinet [5], [6], EtherCAT [7], [8] and Avionics Full Duplex Switched Ethernet (AFDX) [9]) have been very desirable due to the high data bandwidth of up to 1 GB/s and its reliability when operated as a switched, full-duplex network. However, such configuration can be for vehicular producers disadvantageous due to the increased weight of a four wire (two twisted pair) configuration and the added need for specific switches to create such network. A solution to this problem has been recently created, BroadR-Reach [10], which is again a switched network and full-duplex but using only two wires, i.e. a single twisted pair.

Despite this, within recent years, we have been interested in a control network solution using half-duplex Ethernet based on the two wire solution [11], [12]. For this reason, we investigated and developed Media Access Control (MAC)-strategies for half-duplex Ethernet which allow to minimize jitter and delay for control communication. Such strategies are easily implemented on existing Ethernet systems. Using them in the half-duplex frame work reduces weight and cost for switch/router technology (as it can be in principle reduced). In comparison to recent solutions [10], our solution [11], [12] allows complete information sharing across the network (such as CAN) and is rather low cost.

Our approach has been inspired by the CANOpen protocol, where a synchronisation signal triggers nodes connected to the network to send their control signal data packet. In case, several nodes send a data packet, the CSMA/CA-mechanism (Carrier Sense Multiple Access/Collision Avoidance) of the CAN protocol will resolve the order of transmission of these data packets. The synchronization signal combined with the CDMA/CA scheme guarantees data arrival with minimal jitter and delay. Such a scheme within half-duplex Ethernet is not possible due to the inherent CDMA/CD MAC-strategy. However, a global synchronization signal is easily implemented using for instance IEEE 1588 clock synchronization. Thus, it is

in the second step necessary to review and modify the Ethernet MAC (Binary Exponential Backoff, BEB) to avoid packet collisions as much as possible so that transmission delay and jitter following the synchronization signal is minimal. This was achieved in the practical investigations of [11], [12] by providing each transmission node with a specific minimal back-off time. Thus, the basic principle of the suggested communication scheme is to have all nodes sending at a synchronization event, for which then the back-off scheme with specific minimal back-off times minimizes jitter and delay. It was shown in [11], [12] using simulations and practical tests that jitter and delay can be practically kept smaller than with a standard BEB scheme and equal minimal back-off time.

In this paper, a mathematical model is developed for the suggested communication scheme, i.e. this is to provide theoretical evidence for the positive effect of reducing jitter and delay using the suggested MAC. For this reason, a Markov model [13] is derived considering the different states of the back-off scheme, successful transmission, backoff or packet collision. By modelling every single state using a computational approach, it is possible to obtain an overall value for jitter and delay. This is done here for a two node system as an example. Moreover, the back-off scheme is investigated in the context of the suggested communication principles for both, Binary Exponential Backoff and Linear Backoff. Note that for simplicity the network is modelled without any delays in the transmission wiring.

In the next Section, the basic Binary Exponential Back-off scheme is revisited to allow for a better understanding of the suggested control communication approach.

II. HALF DUPLEX ETHERNET REVISED

Before going into the detail of our MAC for control communication we review the basic half duplex Ethernet MAC ideas.

A. Half Duplex Ethernet MAC

The MAC layer of the IEEE 802.3 standard for Ethernet uses the CSMA/CD (Carrier Sense Multiple Access/Collision Detection) protocol with Binary Exponential Backoff [14], [15]. At a time, when a station wants to transmit, it listens to the transmission medium. When a node detects a carrier, its Carrier Sense is turned on and it will defer transmission until the medium is free: if two or more stations simultaneously begin to transmit, a collision occurs. In this case, the BEB algorithm for a random time interval is employed as below:

- When a collision occurs, each CSMA/CD unit chooses to back off for a period of time, determined by the maximal backoff value as an upper limit to the waiting period. Thus, this maximal backoff value is the number of time steps a node will wait at most before trying a retransmission. The first or initial backoff time value is termed '*the minimal backoff time*'. Thus, at the point of a first collision, the maximal back-off value is set to this minimal back-off time.

Each CSMA/CD unit will choose a random backoff time value which follows an equal distribution with an upper bound given by the maximal backoff value.

- Should it happen, that two nodes attempt to retransmit after backoff at the same time, the maximal back-off time value at each node involved in the collision is multiplied by 2 (maximum upper bound of 1024 for the factor). Thus, this causes an exponential growth of the maximal backoff time, i.e. binary exponential backoff.
- On a successful transmission, the transmitting unit sets its backoff value to zero.
- If a unit has attempted backoff 16 times due to collisions for transmitting the same packet, the BEB algorithm forces that unit to discard that packet. Furthermore, the backoff value of this unit is reset to zero, i.e. any new backoff/retransmission attempt will be determined again by the minimal backoff time.

It is evident, that a Linear Backoff scheme is easily employed. In a Linear Backoff scheme, the increase of the maximum backoff window is linear, i.e. $\sim n_f$, on each successive failure, with counter, n_f , in contrast to $\sim 2^{n_f}$ for the BEB scheme. In the next section, an overview of our novel Media Access Control (MAC) strategy is given.

B. The suggested MAC for half-duplex Ethernet

In our work [11], [12], we suggested three MAC principles for real-time control communication. The first principle is the synchronization signal. The second is the introduction of a time slot after the synchronization signal for transmission of each Ethernet packet. Such time slots might be the same for some transmission nodes, i.e. we are reducing with this measure collisions but may not avoid them. This is practical as we are not using high performance switches/router equipment, so that even specific unique time slots may not fully avoid collisions. However, we are not going to use this suggested MAC-principle of time-slots in our analysis to keep things simple. The final principle is the application of different minimal backoff times for each MAC-unit. All data packets have a length sufficiently small, so that they can be transmitted within a sufficiently small time interval¹. However, note that the analysis here is based on the basic idea of time steps (which have for instance a duration of $5.12 \mu s$ in a 100Mbit/s Ethernet network) and does not carry physical units. Thus, the two considered MAC-principles are:

- 1) **Synchronization signal:** Real-time applications require tight synchronization so that the delivery of control messages can be guaranteed within defined message cycle times. Practical implementation is possible by using the IEEE 1588 clock synchronization approach. Thus, a synchronization signal for data transmission is sent at the beginning of a transmission period, where all transmitting nodes are allowed to send a data packet.

¹In our practical papers[11], [12], the length of one data packets was fixed to 48 bytes for a 100Mbps network. Fixed-length transmission has a predictable transmission time and reduces the probability of frame collision.

The time gap between each synchronization signal has to be sufficiently large (i.e. several time steps) to permit transmission of all data packets within the interval defined by the synchronization signal. The CSMA/CD methods have to guarantee that all packet transmissions following a single synchronization signals is carried out with minimal delay and jitter.

- 2) **The backoff scheme** (CSMA/CD) comes into play for the half duplex Ethernet network. Applying different, but fixed minimal backoff times of one or multiple time steps for each transmitting node will allow that some collisions are avoided early on and jitter and delay is kept small.

By assuming an equally distributed probability for collision, transmission and backoff (as explained later), we will now develop a rigorous Markov model for the suggested MAC scheme, which will allow us to show the advantage of the suggested scheme. For this, we provide a short introduction to Markov chains and our model principles first.

III. OVERVIEW OF MARKOV CHAIN MODELLING

Markov chains are an approach in mathematical modelling for random phenomena involving transitions between states with well defined probabilities; i.e. A.A.Markov, 1907 [16] explored chance processes. With this approach, the outcome of a test/experiment can affect the outcome of the next experiment, creating a Markov chain.

In general, a Markov chain considers a process with state X_n at time n , where the state X_n is an integer between zero and N . The state constitutes of what is being modelled, and for the purpose of this overview is left abstract. The one-step transition probability is

$$P(X_{n+1} = b | X_n = a) \quad (1)$$

This probability determines the chance that a process moves from state $X_n = a$, at time step n to state $X_{n+1} = b$ at the later time step $n + 1$.

This allows to create large scale decision processes represented by graphs. Usually, the number of states is finite, i.e. creating a finite sized graph for a process of possibly infinite duration.

A state transition graph for $N = 3$ is shown in Figure 1. For any given state a , Equation (2) must hold;

$$\sum_{k=0}^N P_{ak} = 1 \quad (2)$$

This states that the sum of probabilities $P_{a,b_k} = P(b_k|a)$ of all possible N successor states, b_k , $k = 1, \dots, N$ from a single state, a must be one.

Hence, a finite number of possible outcomes at each stage is required to make sure that we know the probabilities

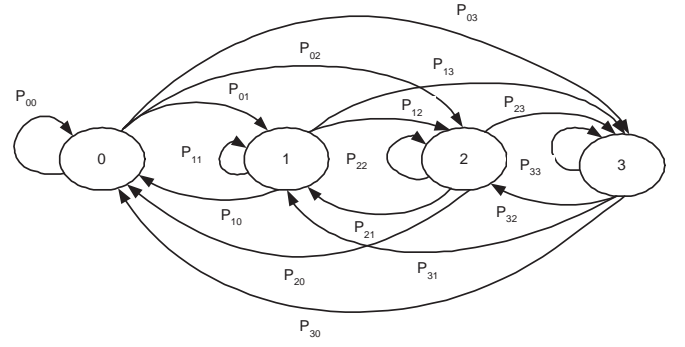


Fig. 1. Graph with four states and Markov chain transition

for any particular outcome at the $j - th$ stage, given the knowledge of the outcome for the first $j - 1$ stages. For each j , a tree graph U_j is obtained [17]. The set paths of this tree serves as a possibility space for any statements relating to the j experiments. Figure 2 illustrates the general concept of our communication approach. Thus, the decision model of a Markov chain is a way of analyzing a set of probabilistic decisions that happen one after another.

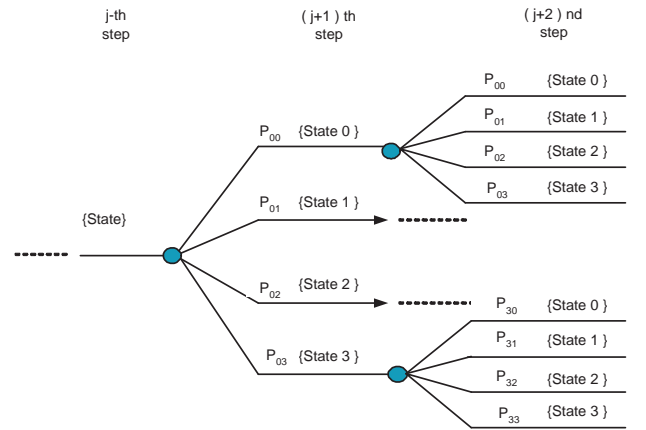


Fig. 2. General tree for the Markov chain transition

For a Markov chain, the large number of decisions, each associated with several outcomes, can result in a very large decision tree. The collection of transition probabilities, for finite Markov Chains, is expressed as a $(N+1)$ -by- $(N+1)$ matrix P called the Nth transition probability matrix, which is shown below:

$$P = \begin{bmatrix} p_{00} & p_{01} & p_{02} & \dots & p_{0N} \\ p_{10} & p_{11} & p_{12} & \dots & p_{1N} \\ p_{20} & p_{21} & p_{22} & \dots & p_{2N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{NN} & p_{NN} & p_{NN} & \dots & p_{NN} \end{bmatrix}.$$

In this case, each row $P_i = [p_{i0}, p_{i1}, p_{i2}, \dots, p_{iN}]$ in P specifies for state i the probabilities of which state $(0, 1, 2, 3, \dots, N)$

is next. This permits the computation of the whole decision tree: $P_j^0 = \text{Prob}[X(0)=j]$ is the initial distribution of states and $P_j^{(n)} = \text{Prob}[X(n)=j]$ gives the probability of being in state j after n steps. Hence, the initial probability vector and the transition matrix P of all transition probabilities completely determine the Markov chain process, e.g. $P_j^{(n)}$. It is sufficient to build the entire tree process. Often the transition matrix P is also a function of the step index, n , i.e. $P = P(n)$.

With this overview, a Markov Chain can be well captured in terms of its dependencies in the system. It also provides a convenient means of specification and automated generation of a large Markov chain.

IV. A PRINCIPAL MARKOV MODEL FOR A TWO-NODE TRANSMITTER SYSTEM

Network performance via the analytical model with a two node transmitter system according to the following cases is investigated:

- Identical minimal backoff time for each node.
- Different minimal backoff times for each node.

Hence, specific minimal backoff times can be assigned to each node.

The proposed Markov model is evaluated by calculating the jitter and delay resulting from the Markov chain tree. The model allows to describe the sending process of two nodes after a single synchronization signal event only. This is sufficient to model the transmission process for all synchronization events as they are from a probabilistic point of view identical. Having explained the transition graph for a specific node, the parameter set is explained which describes the actual Markov model.

1) *Parameters for two-node system:* The parameters involved in this analysis are used to describe the network state and performance. This is best explained for the first considered case, the two node system. The usual starting point is that each node attempts transmission. Once a node has successfully sent a data packet, its task is done. This implies the following states:

- Node 1 sends data : Node 1 successfully sends data without collision.
- Node 2 sends data : Node 2 successfully sends data without collision.
- Node 1 and 2 enter backoff : Following a collision, Node 1 and 2 enter the backoff scheme for a certain backoff time determined by the minimal backoff time and a random multiple of the minimal backoff time. After that waiting/backoff time, the node will attempt retransmission. A maximum retransmission limit can be set. Usually, it is 15 times.
- Node 1 or 2 have unsuccessfully attempted retransmission, i.e. either one of them terminate transmission.

The following vector :

$$\{n, BC_1, BC_2, BD_1, BD_2\} \quad (3)$$

is sufficient to describe the state of the transmission process. Here, n describes the transition step for the Markov chain, i.e. $n = 0, 1, 2, 3, \dots, N$. Thus, the two node backoff process starts at step $n = 0$ and continues to step $n = 1$. BC_1 and BC_2 are the counters in the backoff scheme which count the number of failed attempts for transmission of data packets for each node. Thus, they determine the maximum waiting duration per node. For the Linear Backoff scheme, this is:

$$(BC_1 \cdot BT_1 \quad \text{and} \quad BC_2 \cdot BT_2) \quad (4)$$

where BT_1 and BT_2 are each the minimal backoff times. For the Binary Exponential backoff scheme, the maximum waiting periods are given by:

$$(BT_1 \cdot 2^{BC_1} \quad \text{and} \quad BT_2 \cdot 2^{BC_2}) \quad (5)$$

These values will act as upper bounds for the parameters BD_1 and BD_2 . Thus, the parameters BD_1 and BD_2 determine the current maximal backoff time values of the two nodes. They are chosen according to the maximum counter value, as outlined above. BD_1 and BD_2 are decreased once a node is waiting for the reattempt of transmission, i.e. a single decrement is made at each time step. Once BD_1 and BD_2 are zero the waiting/backoff of the relevant node is finished, i.e. this node will have to send data. Thus, BD_1 and BD_2 act as upper bound in which a node may wait or send data. Note that BC_1 , BC_2 , BD_1 and BD_2 are all free of units, i.e. integers.

This setup allows now to create a Markov chain tree for which the state parameter vector can be explained. As an example, Figure 3 shows the case where both nodes have identical minimal backoff times, $BT_1 = 1$ and $BT_2 = 1$ for the exponential backoff scheme. Note again that the model does not investigate the case of the sending in time slots, i.e. sending of a data packet occurs for all nodes at the occurrence of the synchronization signal followed by the backoff arbitration process if needed. Thus, as in Figure 3, the transmission process starts with:

$$\{0, 0, 0, 1, 1\}$$

This implies at this point $BC_1 = BC_2 = 0$. The backoff delay counters $BD_1 = BD_2 = 1$, will decrease by one which enforces the sending of a data packet. Thus, both nodes will make an attempt to send a data packet at the next step. This certainly will result in a collision, which forces both nodes to enter the backoff mechanism. This creates the state for the exponential backoff scheme:

$$\{1, 1, 1, 2, 2\}$$

At this point, $BC_1 = BC_2 = 2$ will result in the option that each node will either:

- Decrease the backoff counter, i.e. the next step will result in the state $\{2, 1, 1, 1, 1\}$.
- Attempt successfully to send a data packet, i.e. BC_1 , BD_1 or BC_2 , BD_2 are set to zero, while the other will

decrease its counter BC_1 or BC_2 , i.e. $\{2,0,1,0,1\}$ and $\{2,1,0,1,0\}$.

- Cause a collision since both attempt packet transmission. This implies that both nodes will have to increase its attempt counter to $BC_1 = BC_2 = 2$. Thus, for the exponential backoff scheme the state will be $\{2,2,2,4,4\}$.

Each of those options has the same probability of $\frac{1}{4}$.

The next steps follow a similar scheme. For instance, for $\{2,0,1,0,1\}$ and $\{2,1,0,1,0\}$: The transmission has been successful for one of the nodes. Thus, the backoff counter will decrease from 1 to zero for the other node in the third step and cause another successful transmission:

$$\{3,0,0,0,0\}$$

Here, the probability for each of the transitions is 1. Clearly, a complete graphical solution (see Figure 3 for slightly greater detail of the Markov tree) to this Binary Exponential Backoff model is not feasible, for which reason a Matlab based computation can be conducted using the high performance computing facility of the University of Bristol. This will also enable the computation of jitter and delay as explained next.

A. Key characteristic values of the communication system: probabilities, jitter and delay

It is now possible to assess the probability of success and failure of transmission, which builds the basis for computation of jitter and delay. This is again done here for the case of the exponential backoff scheme with identical backoff time $BT_1 = BT_2 = 1$ (see Figure 3). Let us consider first, the probabilities of the initial value:

$$\begin{aligned} P_{x_0} &= (x_0 = \{0,0,0,1,1\}) \\ &= 1 \end{aligned}$$

This is followed by the transition probability:

$$\begin{aligned} P_{1,0,1} &= P(x_{1,1} = \{1,1,1,2,2\} | x_0 = \{0,0,0,1,1\}) \\ &= 1 \end{aligned}$$

In step $n = 2$, the probability for observing a successful transmission, of either of the two nodes, a waiting cycle or a collision is equally distributed. Thus,

$$\begin{aligned} P_{2,1,i} &= P(x_{2,1,i} | x_{1,1} = \{1,1,1,2,2\}) \quad i = 1, 2, 3, 4 \\ &= \frac{1}{4} \end{aligned}$$

where for our example:

$$\begin{aligned} x_{2,1,1} &= \{2,0,1,0,1\} \\ x_{2,1,2} &= \{2,1,0,1,0\} \\ x_{2,1,3} &= \{2,1,1,1,1\} \\ x_{2,1,4} &= \{2,2,2,4,4\} \end{aligned}$$

Thus, in general, for each transition probability of our chain we have:

$$P_{n,f,a} = P(x_{n,a} | x_{n-1,f})$$

where n determines the step instance, $n = 0, 1, 2, 3, \dots, N$, f determines the state at step $(n-1)$ and a determines the state, $a = 0, 1, 2, 3, \dots, N$ at step n . It easily follows:

$$\sum_{a=0,1,2,\dots,N} P_{n,f,a} = 1$$

for fixed value of f .

It is evident that the overall Markov chain is finite, i.e. each node will ultimately enter a state in which it was successful or unsuccessful in transmitting a data packet. Thus, those states which describe both nodes in the final state maybe identified by the transition probabilities:

$$P_{n_{s1},f_{s1},a_{s1}} \quad \text{and} \quad P_{n_{u1},f_{u1},a_{u1}}$$

or

$$P_{n_{s2},f_{s2},a_{s2}} \quad \text{and} \quad P_{n_{u2},f_{u2},a_{u2}}$$

for **Successful** and **Unsuccessful** transmission of either node 1 or 2. There is only a finite number of triples (n_{s1}, f_{s1}, a_{s1}) , (n_{u1}, f_{u1}, a_{u1}) , (n_{s2}, f_{s2}, a_{s2}) and (n_{u2}, f_{u2}, a_{u2})

This implies that the probabilities given by

$$P_{s1} = \sum_{(n_{s1},f_{s1},a_{s1})} P_{n_{s1},f_{s1},a_{s1}} \cdot P_{(n_{s1}-1),f_{s1},\dots} \cdot P_{1,0,1}$$

$$P_{s2} = \sum_{(n_{s2},f_{s2},a_{s2})} P_{n_{s2},f_{s2},a_{s2}} \cdot P_{(n_{s2}-1),f_{s2},\dots} \cdot P_{1,0,1}$$

determines the probability that node 1 and node 2 send out each a packet with success. On the other hand, the probability

$$P_{u1} = \sum_{(n_{u1},f_{u1},a_{u1})} P_{n_{u1},f_{u1},a_{u1}} \cdot P_{(n_{u1}-1),f_{u1},\dots} \cdot P_{1,0,1}$$

$$P_{u2} = \sum_{(n_{u2},f_{u2},a_{u2})} P_{n_{u2},f_{u2},a_{u2}} \cdot P_{(n_{u2}-1),f_{u2},\dots} \cdot P_{1,0,1}$$

determine the probability of ultimately **unsuccessful** transmission. Moreover, $P_{s1} = 1 - P_{u1}$, $P_{s2} = 1 - P_{u2}$. This now also allows for the computation of the (average) delay of a successful transmission:

$$\bar{\tau}_1 = \sum_{(n_{s1},f_{s1},a_{s1})} n_{s1} \cdot P_{n_{s1},f_{s1},a_{s1}} \cdot P_{(n_{s1}-1),f_{s1},\dots} \cdot P_{1,0,1}$$

$$\bar{\tau}_2 = \sum_{(n_{s2},f_{s2},a_{s2})} n_{s2} \cdot P_{n_{s2},f_{s2},a_{s2}} \cdot P_{(n_{s2}-1),f_{s2},\dots} \cdot P_{1,0,1}$$

for each node 1 and 2. The jitter is then defined based on the transmission delay $\bar{\tau}_1$ and $\bar{\tau}_2$ for each of the two nodes:

$$\sigma_{\tau 1} = \sqrt{\sum_{(n_{s1},f_{s1},a_{s1})} (n_{s1} - \bar{\tau}_1)^2 \cdot P_{n_{s1},f_{s1},a_{s1}} \cdot P_{(n_{s1}-1),f_{s1},\dots} \cdot P_{1,0,1}}$$

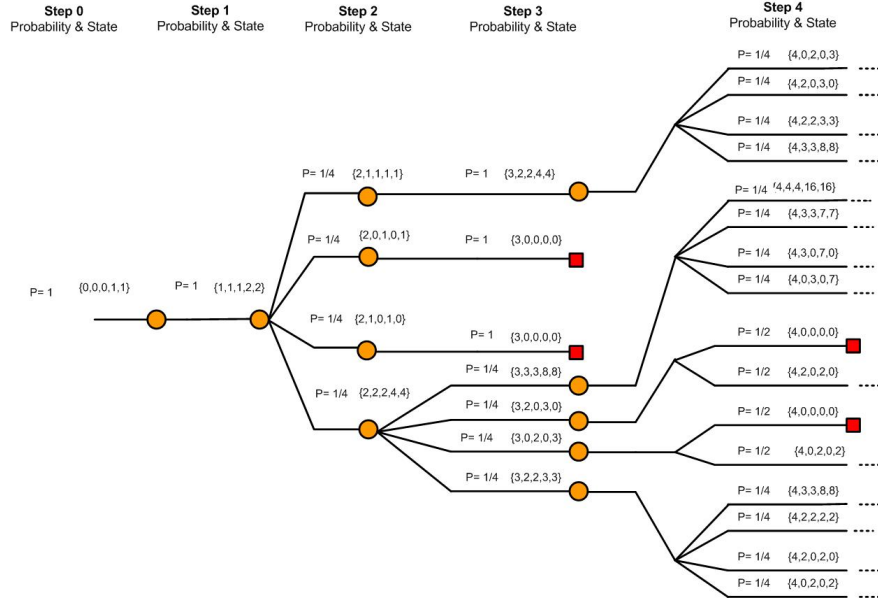


Fig. 3. The two nodes transmitter system for identical backoff time in the Binary Exponential Backoff scheme.

$$\sigma_{\tau 2} = \sqrt{\sum_{(n_{s1}, f_{s1}, a_{s1})} (n_{s2} - \bar{\tau}_2)^2 \cdot P_{n_{s2}, f_{s2}, a_{s2}} \cdot P_{(n_{s2}-1), \dots, f_{s2}, \dots} \cdot \dots \cdot P_{1,0,1}}$$

This allows now for assessment of delay, jitter and probability of transmission success for each of the nodes. The finite Markov state chain is for this reason computed to determine the three values $\bar{\tau}_i, \sigma_i, P_{si}$ for $i = 1, 2$, i.e. for each node.

V. RESULTS

This section discusses performance analysis from the Markov model of the two-node transmitter system. In this case, a Matlab program is used to compute the network states as in Equation (3) of the Ethernet network and to evaluate the performance of the Ethernet communication approach.

A. Identical minimal backoff times

The scenario for Linear and Binary Exponential Backoff is computed, using in particular an overall retransmission attempt limit of 15 times and a minimal backoff time of 1 step for both transmission nodes. In this case, transmission jitter and delay are for both nodes identical. The result for the Linear Backoff scheme shows an advantage over the Binary Exponential Backoff scheme in terms of jitter and delay (Figure 4). This has been so far also confirmed in the simulation experiments of previous work [11], [12]. The performance improvement in terms of jitter and delay for the analytical result is about 25% for the Linear Backoff scheme in relation to the Binary Exponential approach.

B. Different minimal backoff times

In an attempt to enhance the existing standard MAC protocols, different backoff times have been applied for the nodes

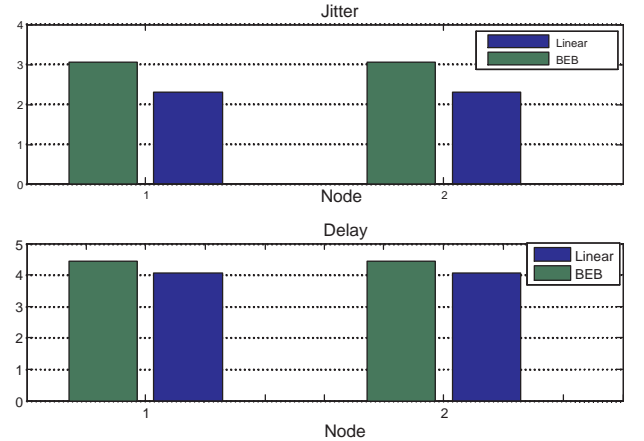


Fig. 4. Result for a two-node transmitter system with identical minimal backoff time.

TABLE I
IDENTICAL MINIMAL BACKOFF TIME FOR A TWO-NODE TRANSMITTER SYSTEM.

Scheme	Jitter		Delay	
	Node 1	Node 2	Node 1	Node 2
Binary Exponential Backoff	3.0158	3.0158	4.4854	4.4854
Linear Backoff	2.2365	2.2365	4.0277	4.0277

in the network system. Here we have chosen $BT_1 = 1$ and $BT_2 = 2$. Results in Figure 5 provide better performance in comparison to the case of identical backoff times.

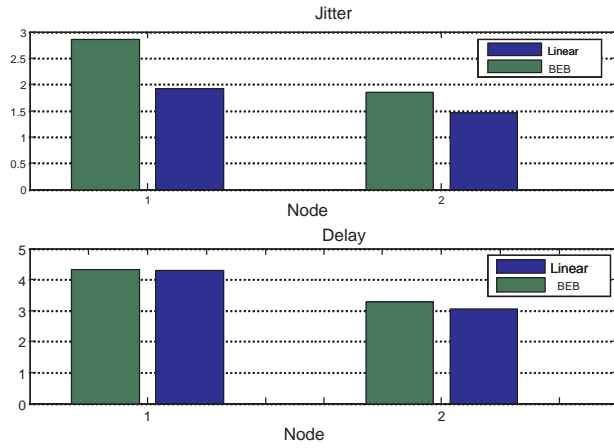


Fig. 5. Result for a two-node transmitter system with different minimal backoff time.

TABLE II
DIFFERENT MINIMAL BACKOFF TIME FOR A TWO-NODE TRANSMITTER SYSTEM.

Scheme	Jitter		Delay	
	Node 1	Node 2	Node 1	Node 2
Binary Exponential Backoff	2.7215	1.7526	4.3205	3.2295
Linear Backoff	1.8021	1.4955	4.2187	3.0781

The node averaged delay for the Binary Exponential Backoff scheme has decreased by at least 15% in comparison to the identical backoff time scheme. In particular, jitter has decreased for both nodes in terms of the values for each node. The improvement of the averaged jitter for the Linear Backoff scheme is 26% while the improvement for the Binary Exponential Backoff scheme is 24%.

Obviously, the results demonstrate that the Ethernet MAC protocol performs reasonably better for the Linear Backoff scheme in comparison to the Binary Exponential Backoff scheme. Moreover, determinism increases by applying different minimal backoff times to each node.

VI. CONCLUSION

In this paper, the developed Markov chain model confirmed the observation from simulation and practical tests [11], [12] for a control communication strategy across half-duplex Ethernet. The two important principles for this communication scheme have been considered in this model: transmission at the synchronisation signal only and different minimal back-off time for each transmission node. A two node system has been considered in a setting of binary exponential back-off and linear back-off. These ideas compared to a standard minimal back-off time of 1 time step, have shown to provide a reduction of at least 24 % in delay and jitter. It is evident that the introduction of time slots after the synchronization event for transmission from each node (of which some time slots are assigned twice or more) will further reduce jitter and delay. Hence, this paper has theoretically confirmed recent practical and simulation

results [11], [12] of these principal ideas of introducing a novel cheap time-triggered communication scheme in half-duplex Ethernet.

ACKNOWLEDGMENT

The authors gratefully acknowledge the contribution of the Ministry of Higher Education Malaysia (MOHE), Universiti Sains Malaysia and the reviewers' comments.

REFERENCES

- [1] "Embedded network," in <http://techon.nikkeibp.co.jp/article>, 1 November 2009.
- [2] G. Kopetz, H. Grünsteidl, "Ttp-a protocol for fault-tolerant real-time systems," in *IEEE Computer*, pp. 14–23, January 1994.
- [3] I. Al Mamoon, A. Muzahidul-Islam, S. Baharun, S. Komaki, and A. Ahmed, "Architecture and communication protocols for cognitive radio network enabled hospital," in *Medical Information and Communication Technology (ISMICT), 2015 9th International Symposium on*, pp. 170–174, March 2015.
- [4] D. Tamas-Selicean and P. Pop, "Optimization of tternet networks to support best-effort traffic," in *Emerging Technology and Factory Automation (ETFA), 2014 IEEE*, pp. 1–4, Sept 2014.
- [5] D. Fontanelli, D. Macii, S. Rinaldi, P. Ferrari, and A. Flammini, "Performance analysis of a clock state estimator for profinet io irt synchronization," in *Instrumentation and Measurement Technology Conference (I2MTC), 2013 IEEE International*, pp. 1828–1833, May 2013.
- [6] V. Sahin, I. Ozcelik, M. Balta, and M. Iskefiyeli, "Topology discovery of profinet networks using wireshark," in *Electronics, Computer and Computation (ICECCO), 2013 International Conference on*, pp. 88–91, Nov 2013.
- [7] T. Maruyama and T. Yamada, "Hardware acceleration architecture for ethercat master controller," May 2012.
- [8] X. Wu, L. Xie, and F. Lim, "Ethercat-enabled next generation baggage handling systems," in *Emerging Technologies Factory Automation (ETFA), 2013 IEEE 18th Conference on*, pp. 1–6, Sept 2013.
- [9] R. Alena, J. Ossenfort, K. Laws, A. Goforth, and F. Figueroa, "Communications for integrated modular avionics," in *Aerospace Conference, 2007 IEEE*, pp. 1–18, March 2007.
- [10] OPENsig, "OPEN Alliance Special Interest Group , accessed 20 July 2013." <http://www.opensig.org>, 2013.
- [11] M. Ishak, G. Herrmann, and M. Pearson, "A simple approach for minimal jitter in ethernet for real-time control communication," in *Advanced Communication Technology (ICACT), 2012 14th International Conference on*, pp. 710–715, Feb 2012.
- [12] M. Ishak, G. Herrmann, and M. Pearson, "Fpga implementation of a simple approach for jitter minimisation in ethernet for real-time control communication," in *High Performance Computing and Communication 2012 IEEE 9th International Conference on Embedded Software and Systems (HPCC-ICSS), 2012 IEEE 14th International Conference on*, pp. 1337–1343, June 2012.
- [13] Z. Q. Chen and M. Fukushima, "Symmetric markov processes, time change, and boundary theory," 2012.
- [14] K. Wang, C. Zhang, X. Ding, S. Ji, and T. Hu, "A new real-time ethernet for numeric control," in *Intelligent Control and Automation (WCICA), 2010 8th World Congress on*, pp. 4137–4141, july 2010.
- [15] J. Talledo, "Design and implementation of an ethernet frame analyzer for high speed networks," in *Electronics, Communications and Computers, 2005. CONIELECOMP 2005. Proceedings. 15th International Conference on*, pp. 171–176, Feb. 2005.
- [16] S. Meyn and L. Richard Tweedie, "Markov chains and stochastic stability," 2009.
- [17] K. Chan, J. Davis, S.-Y. Pai, F. A. Bonilla, J. M. Puck, and M. Apkon, "A markov model to analyze cost-effectiveness of screening for severe combined immunodeficiency (SCID)," *Molecular Genetics and Metabolism*, vol. 104, no. 3, pp. 383 – 389, 2011.